EE247
Lecture 3

• Last week’s summary
• Active Filters
  – Active biquads
    • Sallen- Key & Tow-Thomas
    • Integrator based filters
      – Signal flowgraph concept
      – First order integrator based filter
      – Second order integrator based filter & biquads
  – High order & high Q filters
    • Cascaded biquads
      – Cascaded biquad sensitivity
    • Ladder type filters

Summary
Last Week

• Major success in CMOS technology scaling:
  → Inexpensive DSPs technology
  → Resulted in the need for high performance
     Analog/Digital interface circuitry
• Main Analog/Digital interface building blocks
  includes
  – Analog filters
  – D/A converters
  – A/D converters
Monolithic Filters

- Monolithic inductor in CMOS tech.
  - Integrated L<10nH with Q<10 combined with max. cap. 10pF
  - \( \Rightarrow \) LC filters in the monolithic form feasible: freq>500MHz

- Analog/Digital interface circuitry require fully integrated filters with critical frequencies << 500MHz
- Good alternative:

  \( \Rightarrow \) Active filters built without the need for inductors

2nd Order Transfer Functions (Biquads)

- Biquadratic (2nd order) transfer function:

  \[
  H(s) = \frac{1}{1 + \frac{s}{\omega_p Q_p} + \frac{s^2}{\omega_p^2}}
  \]

  \[
  |H(j\omega)|_{\omega=0} = 1 \quad |H(j\omega)|_{\omega=\infty} = 0 \quad |H(j\omega)|_{\omega=\omega_p} = Q_p
  \]

  Biquad poles @:

  \[
  s = -\frac{\omega_p}{2Q_p} \left(1 \pm \sqrt{1-4Q_p^2}\right)
  \]

  for \( Q_p \leq \frac{1}{2} \) poles are real, complex otherwise
Biquad Complex Poles

\[ Q_p > \frac{1}{2} \]

\[ s = -\frac{\omega_p}{2Q_p} \left( 1 \pm j\sqrt{4Q_p^2 - 1} \right) \]

Distance from origin in s-plane:

\[ d^2 = \left( \frac{\omega_p}{2Q_p} \right)^2 \left( 1 + 4Q_p^2 - 1 \right) \]
\[ = \omega_p^2 \]

s-Plane

radius = \( \omega_p \)

arccos \( \frac{1}{2Q_p} \)

real part = \( -\frac{\omega_p}{2Q_p} \)
Implementation of Biquads

- Passive RC: only real poles - can’t implement complex conjugate poles
- Terminated LC
  - Low power, since it is passive
  - Only noise source → load and source resistance
  - As previously analyzed, not feasible in the monolithic form for \( f < 500 \text{MHz} \)
- Active Biquads
  - Many topologies can be found in filter textbooks!
  - Widely used topologies:
    - Single-opamp biquad: Sallen-Key
    - Multi-opamp biquad: Tow-Thomas
    - Integrator based biquads

Active Biquad
Sallen-Key Low-Pass Filter

- Single gain element
- Can be implemented both in discrete & monolithic form
- “Parasitic sensitive”
- Versions for LPF, HPF, BP, ...
  - Advantage: Only one opamp used
  - Disadvantage: Sensitive to parasitics

\[
H(s) = \frac{G}{1 + \frac{s}{\omega_p Q_p} + \frac{s^2}{\omega_p^2}} \\
\omega_p = \frac{1}{\sqrt{R_1 R_2 C_2}} \\
Q_p = \frac{\omega_p}{1 + \frac{1}{R_1 C_1} + \frac{1 - G}{R_1 C_2}}
\]
Imaginary Axis Zeros

- Sharpen transition band
- "notch out" interference
- High-pass filter (HPF)
- Band-reject filter

\[ H(s) = K \left( \frac{s}{\omega_Z} \right)^2 \left( \frac{s}{\omega_P} \right)^2 \]

\[ H(j\omega) \bigg|_{\omega \to \infty} = K \left( \frac{\omega_P}{\omega_Z} \right)^2 \]

**Note:** Always represent transfer functions as a product of a gain term, poles, and zeros (pairs if complex). Then all coefficients have a physical meaning, reasonable magnitude, and easily checkable unit.

---

Imaginary Zeros

- Zeros substantially sharpen transition band
- At the expense of reduced stop-band attenuation at high frequency

\[ f_p = 100kHz \]
\[ Q_p = 2 \]
\[ f_z = 3f_p \]
Moving the Zeros

\[ f_r = 100kHz \]
\[ Q_r = 2 \]
\[ f_z = f_r \]

Tow-Thomas Active Biquad

Frequency Response

\[ V_{o1} = -k_2 \frac{(b_2a_i - b_i)s + (b_2a_0 - b_0)}{s^2 + a_{i}s + a_0} \]

\[ V_{o2} = \frac{b_2s^2 + b_1s + b_0}{s^2 + a_{i}s + a_0} \]

\[ V_{o3} = -\frac{1}{k_1\sqrt{a_0}} \frac{(b_0 - b_2a_0)s + (a_2b_0 - a_0b_1)}{s^2 + a_{i}s + a_0} \]

- \( V_{o2} \) implements a general biquad section with arbitrary poles and zeros
- \( V_{o1} \) and \( V_{o3} \) realize the same poles but are limited to at most one finite zero

Component Values

\[ b_i = \frac{R_i}{R_kR_iR_kC_iC_i} \]

\[ b_2 = \frac{R_i}{R_k} \]

\[ a_i = \frac{R_i}{R_kR_iR_kC_iC_i} \]

\[ a_1 = \frac{1}{R_kC_i} \]

\[ k_i = \frac{R_iR_iC_iC_i}{R_kR_kC_iC_i} \]

\[ k_1 = \frac{R_i}{R_k} \]

given \( a_i, b_i, C_i \), and \( R_i \)

\[ R_i = \frac{1}{a_iC_i} \]

\[ R_2 = \frac{k_1}{\sqrt{a_0C_i}} \]

\[ R_3 = \frac{1}{k_2\sqrt{a_0C_i}} \]

\[ R_4 = \frac{1}{k_1a_{i}b_1-b_1C_i} \]

\[ R_5 = \frac{k_1\sqrt{a_0}}{k_2C_i} \]

\[ R_6 = \frac{R_i}{b_2} \]

\[ R_7 = k_2R_i \]

it follows that

\[ \omega_r = \frac{R_i}{\sqrt{R_kR_iR_kC_iC_i}} \]

\[ Q_r = \omega_rR_iC_i \]
**Integrator Based Filters**

- Main building block for this category of filters → integrator
- By using signal flowgraph techniques → conventional filter topologies can be converted to integrator based type filters
- Next few pages:
  - Signal flowgraph techniques
  - 1st order integrator based filter
  - 2nd order integrator based filter
  - High order and high Q filters

---

**What is a Signal Flowgraph (SFG)?**

- SFG → Topological network representation consisting of nodes & branches- used to convert one form of network to a more suitable form (e.g. passive RLC filters to integrator based filters)
- Any network described by a set of linear differential equations can be expressed in SFG form.
- For a given network, many different SFGs exists.
- Choice of a particular SFG is based on practical considerations such as type of available components.

What is a Signal Flowgraph (SFG)?

- SFG nodes represent variables ($V$ & $I$ in our case), branches represent transfer functions (we will call these transfer functions branch multiplication factor BMF).
- To convert a network to its SFG form, KCL & KVL is used to derive state space description:
- Example:

Signal Flowgraph (SFG) Rules

- Two parallel branches can be replaced by a single branch with overall BMF equal to sum of two BMFs

- A node with only one incoming branch & one outgoing branch can be replaced by a single branch with BMF equal to the product of the two BMFs

- An intermediate node can be multiplied by a factor ($x$). BMFs for incoming branches have to be multiplied by $x$ and outgoing branches divided by $x$
Signal Flowgraph (SFG) Rules

- Simplifications can often be achieved by shifting or eliminating nodes

A self-loop branch with BMF y can be eliminated by multiplying the BMF of incoming branches by $1/(1-y)$

Integrator Based Filters

1st Order LPF

- Start from RC prototype
- Use KCL & KVL to derive state space description:

$$\frac{V_o}{V_{in}} = \frac{1}{1 + s \cdot RC}$$

- Use state space description to draw signal flowgraph (SFG)
Integrator Based Filters
First Order LPF

• KCL & KVL to derive state space description:

\[ V_C = \frac{I_2}{sC} \]
\[ V_I = V_{in} - V_C \]
\[ I_1 = \frac{V_I}{R_s} \]
\[ I_2 = I_1 \]
\[ V_o = V_C \]

• Use state space description to draw signal flowgraph (SFG)

Normalize

• Since integrators - the main building blocks- require in & out signals in the voltage form (not current)
  
  → Convert all currents to voltages by multiplying current nodes by a scaling resistance \( R^* \)
  
  → Corresponding BMFs should then be scaled accordingly

\[ V_I = V_{in} - V_o \]
\[ I_1 = \frac{V_I}{R_s} \]
\[ V_o = \frac{I_2 R^*}{sC R^*} \]
\[ I_2 = I_1 \]

\[ V_I = V_{in} - V_o \]
\[ I_1 = \frac{R^*}{R_s} V_I \]
\[ V_o = \frac{V_2}{sC R^*} \]
\[ I_2 = I_1 R^* \]
\[ V_2 = V_I \]
Normalize

\[ \begin{align*}
V_{in} & \rightarrow I \rightarrow V_l - I \rightarrow V_o \\
\frac{1}{R_s} & \rightarrow \frac{1}{sC} \rightarrow I \\
I_1 & \rightarrow I \rightarrow I_2
\end{align*} \]

\[ \rightarrow \begin{align*}
V_{in} & \rightarrow I \rightarrow V_l - I \rightarrow V_o \\
\frac{R^*}{R_s} & \rightarrow \frac{1}{sCR^*} \rightarrow I \\
I_1 \times R^* & \rightarrow I_2 \times R^*
\end{align*} \]

\[ \rightarrow \begin{align*}
V_{in} & \rightarrow I \rightarrow V_l - I \rightarrow V_o \\
\frac{R^*}{R_s} & \rightarrow \frac{1}{sCR^*} \rightarrow I \\
V_l & \rightarrow I \rightarrow V_o
\end{align*} \]

Synthesis

\[ \begin{align*}
V_{in} & \rightarrow I \rightarrow V_l - I \rightarrow V_o \\
\frac{R^*}{R_s} & \rightarrow \frac{1}{sC \times R^*} \rightarrow I \\
V_l & \rightarrow I \rightarrow V_o
\end{align*} \]

Consolidate two branches

\[ R^* = R_s \quad \tau = R^* \times C \]

\[ \begin{align*}
V_{in} & \rightarrow I \rightarrow V_l - I \rightarrow V_o \\
\frac{1}{\tau \times s} & \rightarrow I \\
V_l & \rightarrow I \rightarrow V_o
\end{align*} \]
First Order Integrator Based Filter

\[ V_{in} \quad I \quad V_I \quad -I \quad V_o \quad \frac{1}{\tau s} \int \]

\[ H(s) = \frac{1}{\tau s} \]

Opamp-RC Single-Ended Integrator

\[ V_o = -\frac{1}{RC} \int V_{in} \, dt \quad , \quad \frac{V_o}{V_{in}} = -\frac{1}{sRC} \]

\[ \tau = RC \]
1st Order Filter
Built with Opamp-RC Integrator

- Single-ended Opamp-RC integrator has a sign inversion from input to output
  → Convert SFG accordingly by modifying BMF

\[ V_{in}' = -V_{in} \]

1st Order Filter
Built with Opamp-RC Integrator (continued)

\[ \frac{V_o}{V_{in}} = \frac{I}{1+sRC} \]
Opamp-RC 1st Order Filter Noise

Identify noise sources (here it is resistors & opamp)
Find transfer function from each noise source
to the output (opamp noise next page)

\[ \overline{v_n^2} = \sum_{m=1}^{n} \int_{0}^{\infty} |H_m| f |^2 S(f) \, df \]

\[ S(f) \rightarrow \text{Input referred noise spectral density} \]

\[ |H(f)|^2 = \frac{1}{1 + (2\pi fRC)^2} \]

\[ \overline{v_n^2} = \overline{v_{n2}^2} = 4KTR\Delta f \]

\[ \sqrt{\overline{v_o^2}} = \sqrt{\frac{2KTR\Delta f}{\alpha}} \]

\[ \alpha = 2 \]

Typically, \( \alpha \) increases as filter order increases

Opamp-RC Filter Noise
Opamp Contribution

So far only the fundamental noise sources are considered.
In reality, noise associated with the opamp increases the overall noise.
The bandwidth of the opamp affects the opamp noise contribution to the total noise
Integrator Based Filter

2nd Order RLC Filter

- State space description:
  \[ V_R = V_L = V_C = V_o \]
  \[ V_C = \frac{I_C}{sC} \]
  \[ I_R = \frac{V_R}{R} \]
  \[ I_L = \frac{V_L}{sL} \]
  \[ I_C = I_{in} - I_R - I_L \]
  
  *Integrator form*

- Draw signal flowgraph (SFG)

Normalize

- Convert currents to voltages by multiplying all current nodes by the scaling resistance \( R' \)

\[ I_R \quad I \quad I_C \quad I_L \]
\[ \frac{1}{R} \quad \frac{1}{sC} \quad \frac{1}{sL} \]

\[ I_A R' = V_x' \]
\[ V_L \quad I \quad V_C \quad V_R \]
\[ \frac{1}{R} \quad \frac{1}{sC} \quad \frac{1}{sL} \]

\[ R' \quad R \]
\[ V_{i1} \quad V_{i2} \quad V_{i3} \]
\[ V_{o1} \quad V_{o2} \quad V_{o3} \]
Synthesis

\[ V_1 \quad \frac{R^*}{R} \quad I \quad V_2 \quad I \quad V_o \quad \frac{R^*}{sL} \]

\[ V_j \quad -I \quad I \quad V_i \]

\[ \tau_1 = R^* \times C \quad \tau_2 = \frac{L}{R^*} \]

Second Order Integrator Based Filter

Filter Magnitude Response

Normalized Frequency [Hz]

Magnitude (dB)
Second Order Integrator Based Filter

\[
\begin{align*}
V_{BP} &= \frac{\tau_2 s}{\tau_2 s^2 + \beta \tau_2 s + 1} \\
V_{LP} &= \frac{1}{\tau_2 s^2 + \beta \tau_2 s + 1} \\
V_{HP} &= \frac{\tau_2 s^2}{\tau_2 s^2 + \beta \tau_2 s + 1} \\
\tau_1 &= R^* \times C \\
\tau_2 &= L / R^* \\
\beta &= R^*/R \\
\omega_0 &= 1 / \sqrt{\tau_1 \tau_2} = 1 / \sqrt{LC} \\
Q &= 1 / \beta \times \sqrt{\tau_1 / \tau_2}
\end{align*}
\]

From matching point of view desirable:
\[\tau_1 = \tau_2 \rightarrow Q = \frac{R^*}{R^*} \]

Second Order Bandpass Filter Noise

\[\overline{v^2_0} = \sum_{m=1}^{\infty} \int_{-\infty}^{\infty} |H_m(f)|^2 S_d(f) \, df\]

- Find transfer function of each noise source to the output
- Integrate contribution of all noise sources
- Here it is assumed that opamps are noise free (not usually the case!)

\[\overline{v^2_{ni}} = \overline{v^2_{n2}} = 4KT R df\]

Typically, \(\alpha\) increases as filter order increases
Note the noise power is directly proportional to \(Q\)
Second Order Integrator Based Filter
Biquad

- By combining outputs can generate general biquad function:

\[ \frac{V_o}{V_{in}} = \frac{a_1 \tau \omega^2 + a_2 \tau + a_3}{\tau \omega^2 + \beta \tau + 1} \]

s-plane

Summary
Integrator Based Monolithic Filters

- Signal flowgraph techniques utilized to convert RLC networks to integrator based active filters
- Each reactive element (L & C) replaced by an integrator
- Fundamental noise limitation determined by integrating capacitor:
  - For lowpass filter: \( \sqrt{\frac{V_o^2}{V_{in}}} = \sqrt{\frac{kT}{C}} \)
  - Bandpass filter: \( \sqrt{\frac{V_o^2}{V_{in}}} = \sqrt{\frac{\alpha Q kT}{C}} \)

where \( \alpha \) is a function of filter order and topology
Higher Order Filters

- How do we build higher order filters?
  - Cascade of biquads and 1st order sections
    - Each complex conjugate pole built with a biquad and real pole with 1st order section
    - Easy to implement
    - In the case of high order high Q filters \(\Rightarrow\) highly sensitive to component variations
  - Direct conversion of high order ladder type RLC filters
    - SFG techniques used to perform exact conversion of ladder type filters to integrator based filters
    - More complicated conversion process
    - Much less sensitive to component variations compared to cascade of biquads

Higher Order Filters

Cascade of Biquads

Example: LPF filter for CDMA baseband receiver
- LPF with
  - \(f_{\text{pass}} = 650\) kHz \(R_{\text{pass}} = 0.2\) dB
  - \(f_{\text{stop}} = 750\) kHz \(R_{\text{stop}} = 45\) dB
  - Assumption: Can compensate for phase distortion in the digital domain
- 7th order Elliptic Filter
- Implementation with Biquads
  Goal: Maximize dynamic range
  - Pair poles and zeros
    highest Q poles with closest zeros is a good starting point, but not necessarily optimum
  - Ordering:
    Lowest Q poles first is a good start
Filter Frequency Response

Bode Diagram

Phase (deg)

Magnitude (dB)

-80

-60

-40

-20

0

-540

-360

-180

0

3MHz

1MHz

300kHz

Frequency [Hz]

Pole-Zero Map

Q

pole

f

pole

[kHz]

16.7902

659.496

3.6590

611.744

1.1026

473.643

319.568

f

zero

[kHz]

1297.5

836.6

744.0
Biquad Response

Bode Magnitude Diagram

Frequency [Hz]
Intermediate Outputs

<table>
<thead>
<tr>
<th>Frequency [Hz]</th>
<th>Magnitude (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-80</td>
</tr>
<tr>
<td></td>
<td>-60</td>
</tr>
<tr>
<td></td>
<td>-40</td>
</tr>
<tr>
<td></td>
<td>-20</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Magnitude (dB)</th>
<th>LPF1 + Biquad 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>-80</td>
<td></td>
</tr>
<tr>
<td>-60</td>
<td></td>
</tr>
<tr>
<td>-40</td>
<td></td>
</tr>
<tr>
<td>-20</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Frequency [Hz]</th>
<th>Magnitude (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10kHz</td>
<td>10</td>
</tr>
<tr>
<td>100kHz</td>
<td>6</td>
</tr>
<tr>
<td>1MHz</td>
<td>0</td>
</tr>
<tr>
<td>10MHz</td>
<td>-80</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Magnitude (dB)</th>
<th>LPF1 + Biquads 2, 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>-80</td>
<td></td>
</tr>
<tr>
<td>-60</td>
<td></td>
</tr>
<tr>
<td>-40</td>
<td></td>
</tr>
<tr>
<td>-20</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Magnitude (dB)</th>
<th>LPF1 + Biquads 2, 3, 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>-80</td>
<td></td>
</tr>
<tr>
<td>-60</td>
<td></td>
</tr>
<tr>
<td>-40</td>
<td></td>
</tr>
<tr>
<td>-20</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Sensitivity

Component variation in Biquad 4 (highest Q pole):

- Increase $\omega_p4$ by 1%
- Decrease $\omega_z4$ by 1%

High Q poles $\rightarrow$ High sensitivity in Biquad realizations
High Q & High Order Filters

- Cascade of biquads
  - Highly sensitive to component variations → not suitable for implementation of high Q & high order filters
  - Cascade of biquads only used in cases where required Q for all biquads <4 (e.g. filters for disk drives)
- LC ladder filters more appropriate for high Q & high order filters (next topic)
  - Less sensitive to component variations

Ladder Type Filters

- For simplicity, will start with all pole ladder type filters
  - Convert to integrator based form
  - Example shown
- Then will attend to high order ladder type filters incorporating zeros
  - Implement the same 7th order elliptic filter in the form of ladder type
    • Find level of sensitivity to component variations
    • Compare with cascade of biquads
  - Convert to integrator based form utilizing SFG techniques
  - Example shown
LC Ladder Filters

- Made of resistors, inductors, and capacitors
- Doubly terminated or singly terminated (with or w/o $R_s$)

Doubly terminated LC ladder filters $\Rightarrow$ Lowest sensitivity to component variations

LC Ladder Filters

- Design:
  - CAD tools
    - Matlab
    - Spice
  - Filter tables
LC Ladder Filter Design Example

Design a LPF with maximally flat passband:
\[ f_{-3dB} = 10\text{MHz}, \quad f_{\text{stop}} = 20\text{MHz} \]
\[ R_s > 27\text{dB} \]

- Maximally flat passband \( \Leftrightarrow \) Butterworth
- Determine minimum filter order:
  - Use of Matlab
  - or Tables
  - Here tables used

\[ \frac{f_{\text{stop}}}{f_{-3dB}} = 2 \]
\[ R_s > 27\text{dB} \]

Minimum Filter Order
\( \Leftrightarrow \) 5th order Butterworth

From: Williams and Taylor, p. 2-37

---

Find values for \( L \) & \( C \) from Table:
Note \( L \) & \( C \) values normalized to \( \omega_{-3dB} = 1 \)

Denormalization:
Multiply all \( L_{\text{Norm}}, C_{\text{Norm}} \) by:
\[ L = \frac{R}{\omega_{-3dB}} \]
\[ C = \frac{1}{(RX_{\omega_{-3dB}})} \]
\( R \) is the value of the source and termination resistor
(choose both 1\( \Omega \) for now)

Then: \( L = L_{\text{Norm}} \times L \)
\( C = C_{\text{Norm}} \times C \)

From: Williams and Taylor, p. 11.3
LC Ladder Filter Design Example

Find values for L & C from Table:

Normalized values:

|   | C1_norm = C5_norm = 0.618 | C3_norm = 2.0 | L2_norm = L4_norm = 1.618 |

Denormalization:

Since $\omega_{-3dB} = 2\pi \times 10 \text{MHz}$

$L_r = R/\omega_{-3dB} = 15.9 \text{ nH}$

$C_r = 1/(RX_{-3dB}) = 15.9 \text{ nF}$

$R = 1$

$C1 = C5 = 9.836 \text{ nF}$, $C3 = 31.83 \text{ nF}$

$L2 = L4 = 25.75 \text{ nH}$

From: Williams and Taylor, p. 11.3
LC Ladder Filter

Conversion to Integrator Based Active Filter

\[ V_1 = V_{in} - V_2 \]
\[ V_2 = \frac{I_2}{sC_1} \]
\[ V_3 = V_2 - V_4 \]
\[ V_4 = \frac{I_4}{sC_3} \]
\[ V_5 = V_4 - V_6 \]
\[ V_6 = \frac{I_6}{sC_5} \]
\[ V_0 = V_6 \]
\[ I_1 = \frac{V_1}{R_S} \]
\[ I_2 = I_1 - I_3 \]
\[ I_3 = \frac{V_3}{sL_2} \]
\[ I_4 = I_3 - I_5 \]
\[ I_5 = \frac{V_5}{sL_4} \]
\[ I_6 = I_5 - I_7 \]
\[ I_7 = \frac{V_6}{R_L} \]

LC Ladder Filter

Signal Flowgraph

\[ V_1 = V_{in} - V_2 \]
\[ V_2 = \frac{I_2}{sC_1} \]
\[ V_3 = V_2 - V_4 \]
\[ V_4 = \frac{I_4}{sC_3} \]
\[ V_5 = V_4 - V_6 \]
\[ V_6 = \frac{I_6}{sC_5} \]
\[ V_0 = V_6 \]
\[ I_1 = \frac{V_1}{R_S} \]
\[ I_2 = I_1 - I_3 \]
\[ I_3 = \frac{V_3}{sL_2} \]
\[ I_4 = I_3 - I_5 \]
\[ I_5 = \frac{V_5}{sL_4} \]
\[ I_6 = I_5 - I_7 \]
\[ I_7 = \frac{V_6}{R_L} \]
LC Ladder Filter
Signal Flowgraph

$$\begin{align*}
V_{in} & \rightarrow V_1 - 1 \rightarrow V_2 \rightarrow V_3 - 1 \rightarrow V_4 \rightarrow V_5 - 1 \rightarrow V_6 \rightarrow V_{o} \\
I_1 & \rightarrow I_2 - 1 \rightarrow I_3 \rightarrow I_4 - 1 \rightarrow I_5 \rightarrow I_6 - 1 \rightarrow I_7
\end{align*}$$

LC Ladder Filter
Normalize

$$\begin{align*}
V_{in} & \rightarrow V_1 - 1 \rightarrow V_2 \rightarrow V_3 - 1 \rightarrow V_4 \rightarrow V_5 - 1 \rightarrow V_6 \rightarrow V_{o} \\
I_1 & \rightarrow I_2 - 1 \rightarrow I_3 \rightarrow I_4 - 1 \rightarrow I_5 \rightarrow I_6 - 1 \rightarrow I_7
\end{align*}$$

$$\begin{align*}
R^* & \rightarrow \frac{1}{sC_1R} \rightarrow \frac{1}{sL_2} \rightarrow \frac{1}{sC_3} \rightarrow \frac{1}{sL_4} \rightarrow \frac{1}{sC_5} \rightarrow \frac{1}{R_L}
\end{align*}$$
LC Ladder Filter
Synthesize

\[ \begin{align*}
V_{in} & \rightarrow V_1 \rightarrow -1 \rightarrow V_2 \rightarrow 1 \rightarrow V_3 \rightarrow -1 \rightarrow V_4 \rightarrow 1 \rightarrow V_5 \rightarrow -1 \rightarrow V_6 \rightarrow 1 \rightarrow V_7 \\
R^* & \quad \frac{1}{sC_1R^*} \quad \frac{R^*}{sL_2} \quad \frac{1}{sC_3R^*} \quad \frac{R^*}{sL_4} \quad \frac{1}{sC_5R^*} \quad \frac{R^*}{R_L}
\end{align*} \]

\[ V_{in} \rightarrow \frac{R^*}{R_s} \rightarrow V_1 \rightarrow -1 \rightarrow V_2 \rightarrow 1 \rightarrow V_3 \rightarrow -1 \rightarrow V_4 \rightarrow 1 \rightarrow V_5 \rightarrow -1 \rightarrow V_6 \rightarrow 1 \rightarrow V_7 \]

\[ \tau_1 = C_1 R^* \quad , \quad \tau_2 = \frac{L_2}{R^*} = C_2 R^* \quad , \quad \tau_3 = C_3 R^* \quad , \quad \tau_4 = \frac{L_4}{R^*} = C_4 R^* \quad , \quad \tau_5 = C_5 R^* \]

Building Block:
RC Integrator

\[ \frac{V_2}{V_1} = -\frac{1}{sRC^*} \]
Negative Resistors

Synthesize
Frequency Response

Scale Node Voltages
Scale $V_o$ by factor “s”
Total noise: 1.4 μV rms (noiseless opamps)

That’s excellent, but the capacitors are very large (and the resistors small). Not possible to integrate.

Suppose our application allows higher noise in the order of 140 μV rms...

**Scale to Meet Noise Target**

Scale capacitors and resistors to meet noise objective

\[ s = 10^{-4} \]

Noise: 141 μV rms (noiseless opamps)
Completed Design

Sensitivity

• $C_1$ made (arbitrarily) 50% (!) larger than its nominal value
• 0.5 dB error at band edge
• 3.5 dB error in stopband
• Looks like very low sensitivity